Document: GWD-Perf-15-1
Category: Recommendation
Expires: December 2002

Dan Gunter
Lawrence Berkeley National Laboratory
Brian Tierney
Lawrence Berkeley National Laboratory
June 2001

ASCII Timestamp Format for Grid Computing

1.0 Introduction

There are many occasions for producing and consuming timestamps in computing, ranging from performance analysis to security protocols. Grid applications and protocols which produce and consume timestamps need to interoperate, either in real-time or by storing and using information in directory services or archives. A standard for timestamp representation needs to be defined which satisfies these requirements. This representation must be composed of both a model and a format. In the case of Grid timestamp representation, the model and format(s) have been separated into separate documents. This document addresses a specific timestamp format; the underlying model, including a definition of the terms *resolution* and *accuracy,* is presented in [4].

This document describes a timestamp format composed entirely of printable characters from the American Standard Code for Information Interchange (ASCII) [1]. An ASCII timestamp format is useful primarily because it is plain, readable, text. This allows humans to read and compose "raw" timestamps, and it also allows programs to manipulate timestamps with standard string functions. Additionally, some commonly used application envelopes, for example LDIF [3] and anything based on XML [2], are not optimized for binary data. In these cases, and especially when the volume and frequency of timestamps will not pose a significant load on the system, an ASCII timestamp may be appropriate.

2.0 ASCII Timestamp Format

There are three parts to the timestamp format, which correspond to the parts of the timestamp model (see [4]): value, resolution, and accuracy. Following the timestamp model, all three parts of the timestamp value are required. In this section, each part of the format is first discussed separately, then all three parts are combined in a BNF grammar and a series of examples.

2.0.1 Value

There are two good candidates for an ASCII timestamp format. The first contains separators between the fields, for example: 2000-10-26T08:34:26.30323. The second format is one where the year, month, day, etc. are a single concatenated string, for example the NetLogger [6] date format: 20001026083426.30323. Although the latter is slightly easier for a computer to generate and parse, the primary purpose of this timestamp format is readability, and the former representation allows for easier visual inspection.

Thus, we recommend a Grid timestamp value format that is a highly readable variation from the ISO8601 time standard [5]:

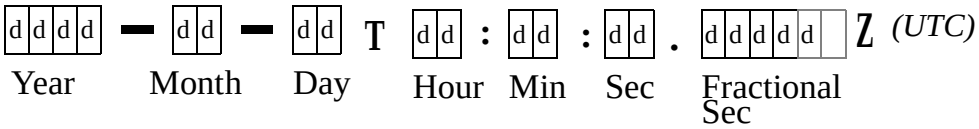

Figure 1: ASCII timestamp value representation

For example, 8:34am, 26 seconds, and 350 milliseconds on October 26th, 2000 UTC would be formatted as: “2000-10-26T08:34:26.350Z”. The “Z” at the end signifies “UTC”, and is required. The length of the fractional seconds after the decimal point is from 1 to 10 digits, with common lengths expected to be (but not limited to) 3, 6, and 9 digits.

2.0.2 Resolution

Resolution is indicated in seconds. Typical values will be 1 (second), .001 (milliseconds), .000001 (microseconds) and .000000001 (nanoseconds). The letter “r” must immediately follow the value.

For example, a timestamp with a resolution of 10 seconds would read: “2000-10-26T08:34:26Z10r”.

If no resolution information is available, the letter “r” should be used with no preceding digits.

2.0.3 Accuracy

The decimal number for the accuracy is placed before the letter “a”, and may have a decimal point and up to 10 digits on either side:

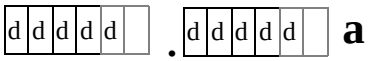

Figure 2: ASCII timestamp accuracy representation

If no accuracy information is available, the letter “a” should be used with no preceding digits.

2.0.4 Backus-Naur Form (BNF) Grammar

Following is a BNF grammar for the ASCII timestamp format. When a comment states “between X and Y”, it means the range X toY, inclusive of X andY. The notation “symbol{a,b}” is used to indicate that the symbol is repeated from *a* to *b* times, inclusive of *a* and *b*.

1. Timestamp ::= value resolution accuracy
2. value ::= year ‘-’ month ‘-’ day ‘T’ hour ‘:’ minute ‘:’ second [‘.’ fraction] ‘Z’
3. resolution ::= [float] ‘r’
4. accuracy ::= [float] ‘a’
5. year ::= d d d d
6. month ::= d d (between 01 and 12)
7. day ::= d d (between 01 and 28-31, depending on month/year)
8. hour ::= d d (between 00 and 23)
9. minute ::= d d (between 00 and 59)
10. second ::= d d (between 00 and 59)
11. number ::= dno0 d{0,9}
12. fraction ::= d{1,10}
13. float ::= number ‘.’ fraction | number | ‘.’ fraction
14. d ::= ‘0’ | ‘1’ | ‘2’ | ‘3’ | ‘4’ | ‘5’ | ‘6’ | ‘7’ | ‘8’ | ‘9’

15.dno0 ::= ‘1’ | ‘2’ | ‘3’ | ‘4’ | ‘5’ | ‘6’ | ‘7’ | ‘8’ | ‘9’

2.0.5 Examples

We now illustrate the ASCII timestamp format with a few examples:

- October 26, 2000 at 8:34am and 26 seconds, with a resolution of 1 millisecond and an accuracy of plus or minus one half of a second, would be represented as:

  2000-10-26T08:34:26Z.001r.5a

- January 1, 2001 at 3:12pm and 5 seconds, with a resolution of 5 seconds and an accuracy of plus or minus 10 minutes, would be represented as:

  2001-01-01T15:12:05Z5r600a

- August 26, 1970 at 12:00 noon and 20 seconds and 356,675 microseconds, with a resolution of 1 nanosecond and an accuracy of plus or minus 10 microseconds, would be represented as:

  1970-08-26T12:00:20.356675Z.000000001r.00001a

- October 26, 2000 at 8:34am and 26 seconds, with no resolution or accuracy specified would be represented as:

  2000-10-26T08:34:26Zra

3.0 Summary

This document has presented an ASCII format for representing timestamp information. This format allows for, but does not require, resolution and accuracy information. All times are represented in UTC format, so conversions may be necessary to and from local time.

4.0 References

[1] ANSI x3.4, “American Standard Code for Information Interchange”

[2] Extensible Markup Language (XML), World Wide Web Consortium, http://www.w3.org/xml/

[3] G. Good, “The LDAP Data Interchange Format - Technical Specification”, IETF RFC 2849, June 2000.

[4] D. Gunter, B. Tierney, “A Timestamp Model for Grid Computing”, GWD-Perf-14-1 (non-normative). http://www-didc.lbl.gov/GridPerf/papers/GWD-GP-14-1.pdf

[5] ISO-8601, “Data Elements and Interchange Formats - Information Exchange - Representation of Dates and Times”, International Organization for Standardization, 1888 http://www.iso.ch/markete/8601.pdf

[6] Tierney, B., W. Johnston, B. Crowley, G. Hoo, C. Brooks, D. Gunter. “The NetLogger Methodology for High Performance Distributed Systems Performance Analysis”, Proceeding of IEEE High Performance Distributed Computing conference (HPDC-7), July 1998, LBNL-42611.